Security of Internet Location Management

Tuomas Aura and Michael Roe
Microsoft Research
7 J J Thomson Avenue, Cambridge CB3 0FB, UK
{tuomaura,mroe}@microsoft.com

Jari Arkko
Ericsson Research NomadicLab
LM Ericsson, FIN-02420 Jorvas, Finland
jari.arkko@ericsson.com

Abstract

In the Mobile IPv6 protocol, the mobile node sends binding updates to its correspondents to inform them about its current location. It is well-known that the origin of this location information must be authenticated. This paper discusses several threats created by location management that go beyond unauthentic location data. In particular, the attacker can redirect data to bomb third parties and induce unnecessary authentication. We introduce and analyze protection mechanisms with focus on ones that work for all Internet nodes and do not need a PKI or other new security infrastructure. Our threat analysis and assessment of the defense mechanisms formed the basis for the design of a secure location management protocol for Mobile IPv6. Many of the same threats should be considered when designing any location management mechanism for open networks.

1. Introduction

This paper describes attacks against mobile and stationary Internet nodes by exploiting location management features of the Mobile IPv6 (MIPv6) protocol and other Internet mobility protocols. We analyze the threats created by location management and the advantages and limitations of various security mechanisms.

It is well known that false location information can corrupt directories and routing tables, leading to misrouting of confidential information, highjacking of connections, and denial of service (DoS) because honest nodes cannot communicate. Cryptographic authentication of the location information is usually seen as a key defense mechanism. For example, authentication of updates to location information is mandatory in Mobile IPv4 [PJ01] and Mobile IPv6 [JPA02], and in Dynamic DNS [VTRB97]. This paper reports the lessons we learned when we set out to design such an authentication protocol for MIPv6.

We first overview known attacks that use unauthentic location data (Section 2) and discuss authentication mechanisms (Section 3). We are particularly interested in mechanisms that allow authentication between arbitrary Internet nodes without prior trust relationships, public-key infrastructure (PKI) or trusted third parties. We then present new types of attacks that go beyond unauthentic messages. In particular, we explain how even strongly authenticated location management can be exploited in DoS attacks. First, data flows can be redirected to flood third parties who are not taking part in the mobility protocol (Section 4). Second, the attacker may exploit features of a location management protocol to exhaust the resources of either the mobile or the correspondent, for example, by inducing unnecessary authentication (Sections 5 and 6). Finally, we make some notes about prioritizing security goals and combining multiple levels of authentication (Section 7).

It is essential to understand that some of the threats may be acceptable or too expensive to prevent completely. Different security mechanisms provide variable levels of guarantees for variable security properties at variable cost. The challenge is to find an acceptable level of protection at an acceptable price. It is not our goal to create a general infrastructure for strong authentication. Instead, the aim is to make sure that the introduction of a new technology, mobility, does not expose the current Internet to uncontrolled threats. Therefore, we can resort to some relatively weak and inexpensive security mechanisms that nevertheless solve the problems at hand.

The ideas presented in this paper formed the basis for the design of the secure location management protocol in the current Mobile IPv6 specification [JPA02]. We believe that the same threats and defenses should be considered in the development of any location management protocol for open networks.

1.1. The Mobile IPv6 Protocol

This section gives a brief overview of mobility and the Mobile IPv6 architecture. We avoid using protocol-specific terms whenever possible.

IP mobility means that an Internet node moves from one location, i.e. IP address, to another, either because it

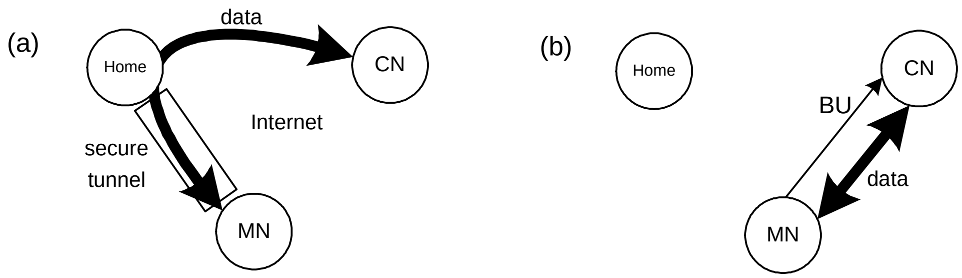

Figure 1. Mobile IPv6 packet forwarding (a) and binding update (b)

moves physically between network coverage areas or media types, or because its logical point of network access changes. The main goals of mobility protocols are to maintain existing connections over location changes and to ensure that the mobile can be reached at its new location. With *location management* we mean any mechanism for informing other nodes about the mobile's current address. Typically, location management either utilizes some kind of directory service where the mobile's location is maintained or it involves direct notifications to the nodes that need to know about the new location.

In Mobile IPv6, every *mobile node (MN)* has a home network and is identified by a *home IP address* on that network. The 128-bit IPv6 address consists of a 64-bit *routing prefix*, which is used for routing the packets to the right network, and a 64-bit *interface identifier*, which identifies the specific node on the network and can essentially be a random number. Thus, IP addresses in MIPv6 can identify either a node or a location on the network, or both.

A router called *home agent* at the home network acts as the mobile's trusted agent and forwards IP packets between the mobile's *correspondent nodes (CN)* and its current location, called *care-of address* (Figure 1(a)). The MIPv6 protocol also includes a location management mechanism called *binding update (BU)*. When the mobile changes its current address, it can send BUs to its correspondents and home agent to notify them about the new location so that they can communicate directly (Figure 1(b)). The mobile may also be triggered to sending a BU when it receives a packet from a new correspondent via the home agent.

The mobile node and its home agent have a permanent trust relationship and a preconfigured security association for encrypted and authenticated communication. The mobile informs the home agent about its location via this secure tunnel. We assume these messages to arrive safely and have nothing further to say about them. The mobile and its home agent can cooperate to send BUs to the correspondents, with which they often have no pre-existing relationship. The correspondent stores the location information in a *binding cache entry*, which needs to be refreshed regularly by sending a new BU. The topic of this paper is the security of these BUs, which are sent to arbitrary correspondents.

It should be noted that there are several alternative approaches to Internet mobility and the attacks and protection mechanism identified in this paper are general enough to be applicable to many such mechanisms. It is beyond the scope of this work to compare the relative merits of the alternative mobility protocols.

Another issue that we do not address is location privacy. MIPv6 does nothing special to try to hide the mobile's home address or current location from others. Nevertheless, the protocol is relatively privacy-friendly: the mobile's current location is tracked by its own home agent but not by any global or public directory, the mobile is free to use temporary and multiple home addresses, and sending BUs to correspondents is a voluntary optimization for the mobile.

2. Unauthentic Location Information

This section gives a brief overview of the threats that result from not authenticating location information. Readers familiar with the topic may want to skip to Section 3.

Unauthenticated location information makes it possible for an attacker to misinform correspondents about the mobile's location and, thus, to redirect packets intended for the mobile to a wrong destination. This can lead to the compromise of secrecy and integrity as well as denial-of-service because the target nodes are unable to communicate.

When sending false BUs, the attacker can use its own address as the false care-of address and pretend to be the mobile. This way, it can highjack existing connections between the mobile and its correspondents or open new ones. The attacker can also redirect the packets to a random or non-existent care-of address in order to disrupt communication with the mobile. It has to send a new binding update every few minutes to refresh the binding cache entry at the correspondent.

These attacks are alarming because the attacker can be anywhere on the network and all Internet nodes are potential targets. All IPv6 nodes must support the correspondent functionality and the addresses of mobile nodes are indistinguishable from those of stationary ones. Thus, the attacker can make any node believe that any other node, even a non-mobile one, is mobile and has moved to the false care-of address. This is, to some extent, a side effect of the effort to make mobility transparent.

In order to send false BUs, the attacker needs to know the IP addresses of both the communicating nodes. This means that nodes with have well-known and permanent addresses, such as public servers and those using stateless auto-configuration [TN98], are most vulnerable. They include nodes that are a part of the network infrastructure, such as DNS servers, which are particularly interesting targets for DoS attacks. Frequently changing random addresses, e.g. ones created by IPv6 addressing privacy features [ND01], may mitigate the risks to some extent.

Obviously, end-to-end encryption and integrity protection of payload data, e.g. with authenticated SSL or IPSec, can prevent the attacks against data secrecy and integrity but not denial-of service. Two stationary nodes that know each other to be stationary could be configured to refuse BUs from each other.

We have considered only active attackers because in order to redirect packets, the attacker must sooner or later send one or more messages. In fact, the active attacks are easier for the average attacker than passive ones would be. In most active attacks, the attacker can initiate the BU protocol execution at any time while passive attacks would require the attacker to wait for suitable messages to be sent by the target nodes.

3. Authentication of Location Data

This section discusses proposed authentication methods for location information. The two first techniques (Sections 3.1-3.2) are relatively strong and involve public-key algorithms. Section 3.3 presents a relatively weak routing-based authentication methods that would be labeled as insecure in traditional network security thinking. Nevertheless, it provides a well-defined level of assurance in the real networks and can complement or even replace the stronger methods. Instead of trying to prevent all attacks, the best strategy is often to limit the number of potential attackers that can attack a particular target, and to reduce the number of targets a potential attacker can threaten.

Any authentication protocol has to take into account replay attacks. A nonce-based freshness mechanism seems practical because the authentication and DoS protection mechanisms described in Sections 3.3 and 4.2 use nonces anyway. Time stamps would be problematic because mobile nodes may not be able to maintain sufficiently accurate clocks. Sequence-numbered BUs, on the other hand, could be intercepted and delayed for later attacks.

3.1. Public Key Authentication

An obvious solution to the authentication of location information would be to use a suite of strong generic authentication mechanisms and a trust infrastructure, such as IPSec, IKE and a X.509-based PKI. There are, however, reasons why the generic protocol suites may not be good for the purpose. First, the generic authentication protocols have usually been designed with general-purpose computers and application-level security requirements in mind. The overhead of these protocols can be too high for low-end mobile devices and for a network-layer signaling protocol. Second, an Internet mobility protocol should allow anyone to become mobile and it must allow all Internet nodes as correspondents. This means that a single PKI should cover the entire Internet, which is a formidable goal when even local infrastructures have failed to emerge at the expected rate. Therefore, it is necessary to look for alternative solutions that do not rely on such global infrastructure.

There are nevertheless situations where it is possible, and advisable, to apply the strong generic authentication solutions. In closed user groups and high-security environments, it may be possible to set up a PKI and to require location information to be strongly authenticated between the group members.

3.2. Cryptographically Generated Addresses

A recently discovered technique provides an intermediate level of security below strong public-key authentication and above routing-based methods (which will be described in the following section). The idea, first introduced in a MIPv6 BU authentication protocol called CAM [OR01], is to form the last 64 bits of the IP address (the interface identifier) by computing a 64-bit one-way hash of the node's public signature key. The node signs its location information with the corresponding private key and sends the public key along with the signed data. The recipient hashes the public key and compares the hash to the address before verifying the signature on the location data. This prevents anyone except the node itself from sending location updates for its address. The attraction of this technique is that it provides public-key authentication of the IP address without any trusted third parties, PKI, or other global infrastructure. Several other BU authentication protocols have been proposed based on this idea [Nik01, MC02, RAOA02].

The main weakness of the scheme is that at most 64 bits of the IP address can be used for the hash. It is imaginable that a brute force attack would become possible during the lifetime of the IPv6 protocol.

Generating strong signature keys is expensive but there may be relatively fast ways of generating weak signature keys, which the correspondent may not be able to tell apart from strong ones. Advances in storage technology may enable the attacker to create a large enough database for finding matching keys at high probability. In order to make such brute-force attacks less attractive, we suggest including the routing prefix of the network into the input of the hash function:

Interface Id = Hash64(Public Key | Routing Prefix | ...)

This forces the attacker to perform the search separately for each prefix. Generating new public keys and changing addresses at regular intervals could also discourage brute-force attacks against individual nodes.

Another limitation of the cryptographically generated addresses (CGA) is that although they prevent the theft of another node's address, they do not stop the attacker from inventing new false addresses with an arbitrary routing prefix. The attacker can generate a public key and a matching IP address in any network. This means, as we will see below, that CGA addresses prevent some packet-flooding attacks against individual addresses but not against entire networks. Moreover, public-key protocols (both PKI-based and CGA-based ones) are computationally intensive and therefore expose the participants to denial-of-service attacks (see Sections 5.1-5.2).

3.3. Assuming a Safe Route

Another way to create a secure connection where none exists is to send the first message through a relatively safe route and hope that it is not intercepted. In MIPv6, the most reasonable assumption is to trust the route between the correspondent and the mobile's home agent. The correspondent can send a secret key in plaintext to the mobile along this path. The mobile's agent then forwards the key through a secure tunnel to the mobile, which uses it for authenticating a binding update to the correspondent. The message flow of such a *routing-based authentication* protocol is shown in Figure 2(a). The last message contains the BU and a message authentication code (MAC) computed with the secret key.

The assumption may be reasonable because very few Internet nodes can listen to or modify packets on the right routers to mount an attack against a given connection. Even an attacker in control of some routers can only interfere with a limited number of connections because most Internet traffic will not be routed through the compromised routers. Typically, at most few dozen routers see the secret keys for a specific connection and thus are able to redirect it. Although not secure in the classical sense, this is a vast improvement compared to the completely unauthenticated situation where any Internet node can attack any other nodes. Moreover, the home agent and the correspondent are typically located on the wired network and their communication is relatively secure compared to the packets to and from a wireless mobile.

The assumption can also be justified by the fact that an attacker on the route between two stationary nodes (a mobile at home and a correspondent) can mount equally damaging attacks against the communication between them. (Recall that our goal was to address the additional threats created by mobility, not ones that exist in the current Internet.)

The routing-based authentication is somewhat weaker than the CGA-based protocols or ones that combine both methods. It may be the best choice when cryptographically generated addresses are not available or public-key cryptography is considered as too expensive. It is important to understand that the key K in Figure 2(a) must not be used for general authentication but only for protecting location information sent by the mobile to the correspondent.

3.4. Some Failed Ideas

Some proposals for BU authentication depended on sending two pieces of authentication data between the correspondent and the mobile via two independent routes and hoping that most attackers are unable to capture both of them. Unfortunately, these protocols did not provide any more security than our idea of sending a single value along a single route. The reason is that although the routes to an honest mobile usually form a triangle with two independent paths, a false mobile (i.e. the attacker) may be located so that the routes overlap and the attacker can control all communication from a single location.

Another idea is the so called leap-of-faith authentication where the mobile sends a session key insecurely to the correspondent at the beginning of their correspondence and the key is used to authenticate subsequent BUs. This does not work (unless they key is sent in a way that takes advantage of a relatively safe route) because the attacker can send its false key before the authentic mobile sends the authentic key. Furthermore, there must be a recovery mechanism for situations where the mobile or the correspondent loses its state, and the attacker can exploit this mechanism.

The leap-of-faith authentication is suitable for situations where a human user, or some other factor outside the attacker's control, at random times initiates the protocol execution. The party making the leap must always be the one that initiates the protocol. In such situations, it may be reasonable to argue that an attacker is unlikely to be present at the time of the unauthenticated key exchange. In BU authentication, the protocol is usually initiated by the mobile but the leap in faith should be made by the correspondent. Also, the attacker can trigger the BU protocol at any time by sending to the mobile's home address a spoofed packet that appears to come from the correspondent.

3.5. The Role of Ingress Filtering

Ingress filtering [FS00] is another way of limiting the number of potential attackers and their targets. Ingress filtering means that a gateway router or firewall at the boundary of a network checks the source addresses of all outgoing packets and drops ones that do not originate from the network. This prevents nodes on the network from sending spoofed packets that appear to come from other networks.

Since the mobile's new address in a MIPv6 binding update is usually sent in the source address field of the IP packet header, ingress filtering seems to limit the choice of false addresses. There are, however, two well-known weaknesses in this thinking. Firsts, ingress filtering must be applied on the attacker's local network; on the target network it makes no difference. Second, the MIPv6 draft standard specifies an alternative mechanism (Alternative Care-of Address sub-option) that can be used for sending a false care-of address without source spoofing. While it is advisable to apply ingress filtering in as many networks as possible, one cannot rely on this to stop all attacks.

4. Bombing Attacks and Stopping Them

The authentication mechanisms discussed above provide varying levels of assurance that the location information originates from the authentic mobile or its trusted agent. On the other hand, they do nothing to keep the mobile from lying about its location. That is, an attacker may be able to give a false care-of address to the correspondent and thus redirect data to this address. We explain the attacks in Section 4.1 and suggest defenses in the following subsections.

4.1. Redirecting Unwanted Data

The false Binding Updates (Section 2) could be used for amplifying packet-flooding DoS attacks. If the attacker knows that there is a heavy data stream between two nodes, it can redirect the stream to the target. Obviously, BU authentication prevents this straightforward attack.

But authentication does not prevent the attacker from lying about its own location. If the attacker acts itself as the mobile, it can send false location data to its correspondents and get them to send traffic to an arbitrary IP address. It first subscribes to a data stream (e.g. a video stream from a public web site) and then redirects this to the target address. This technique can be used to bomb any Internet node with excessive amounts of data. The attacker can also target a network by redirecting data to one or more IP addresses within the network.

The attacker may even be able to spoof the acknowledgements. For example, consider a TCP stream. The attacker performs the TCP handshake itself and thus knows the initial sequence numbers. After redirecting the data to the target, it suffices to send one spoofed acknowledgment per TCP window to the correspondent. (Actually, TCP provides some protection against this attack: If the target address belongs to a real node, it will respond with TCP Reset, which prompts the correspondent to close the connection. On the other hand, if the target is a non-existent address, the target network may send ICMP Destination Unreachable messages. Not all networks send this latter kind of error messages, and some correspondents may ignore them because they are also receiving the spoofed acknowledgments. Other transport-layer protocols may behave less gracefully.)

The attack is not specific to MIPv6. For example, when dynamic updates are made to Secure DNS, there is no requirement or mechanism for verifying that the registered IP addresses are true [Eas97]. ICMP Redirect [NNS98] messages enable a similar attack on the scale of a local network. We expect there to be other protocols with the same type of vulnerability.

A variation of the bombing attack targets the home network instead of the care-of address. This attack is specific to mobility protocols like MIPv6 where the mobile has a default address, to which data will be sent when its current location is unknown. The attacker claims to have a home address in the target network. It starts downloading a data stream and either sends a request to delete the binding cache entry or allows it to expire. This redirects the data stream to the false home address. BU authentication usually limits the attacker's choice of targets but care must be taken when designing the protocol. For example, CGA-based protocols prevent targeting of individual addresses but allow the attacker to bomb a network by generating a new address with its routing prefix.

The attacks are serious because the target can be any node or network, not only a mobile one. What makes them particularly serious compared to the other attacks is that the target itself cannot do anything to prevent the attack. For example, it does not help if the target stops sending or accepting binding updates. The damage is worst if these techniques are used to amplify the effect of a distributed denial of service (DDoS) attack.

The attacker needs to find a correspondent that is willing to send data streams to unauthenticated recipients. Many popular web sites provide such streams. The attacker also needs to know the target's IP address but it may attack an entire network by redirecting data to a nonexistent address and congesting the link toward the network. In some cases, a firewall on the border of the target network may be able to filter out packets to nonexistent addresses. However, IPv6 addressing privacy features [ND01] make such filtering difficult.

4.2. Return Routability

The most effective way to limit opportunities for bombing attacks is to test the return routability (RR) of the mobile's new address. That is, the correspondent sends a packet with a secret value to the new location and accepts the binding update only if the mobile is able to return the value (or its hash). This proves that the mobile can receive packets at the address where it claims to be. Some malicious entities (e.g. ones on the correspondent's local network) may be able to capture a test packet but the number of potential attackers is dramatically reduced.

Figure 2(b) shows how a BU is authenticated using two secrets, which the correspondent sends to the mobile's home and care-of addresses. The secret *Kb* sent directly to the care-of address forms the RR test. The RR test can be seen as a variation of the cookie exchange, which has been used as part of the TCP handshake [SKK+97] and in authentication protocols, including Photuris [KS99].

In MIPv6, the expiry of a binding cache entry poses a special problem. Deleting the cache entry effectively means that the mobile's new address defaults to the home address, but since the mobile may have become unreachable, it is not always possible to test RR for the new address. One solution would be to mark the cache entry as invalid and to stop sending data to the mobile until the RR test succeeds. This could, however, mean that some cache entries are never deleted. Instead, we suggest doing an additional RR test for the home address during every binding update so that when the cache entry needs to be deleted for any reason (e.g. BU cancellation, expiring cache entry, or failing BU authentication), a successful RR test for the home address has always been performed recently and the cache entry can be deleted immediately. This limits bombing-attack targets to networks where attacker has recently visited.

In routing-based authentication (Section 3.3) where the correspondent sends a plaintext key to the mobile via its home address, the same secret key can also serve as the RR test for the home address. Thus, the correspondent in Figure 2(b) does test return routability of both the home and care-of addresses.

It is also important to note that the return routability test is complementary to CGA-based BU authentication, which does not prevent bombing of the home network.

4.3. Relation to Flow Control

It can be argued that the bombing attacks are a flow-control issue and therefore should be taken care of in the transport layer rather than in the IP layer. That is, when sending a data flow into a new route, the correspondent should first verify that this route can accept the data. It could start by sending a single packet and gradually increase the transmission rate. For TCP streams, the natural solution would be to reset the TCP window size to one packet when the mobile moves. This would, in effect, test return routability of the new route before sending large amounts of data into it.

Unfortunately, adding a secure RR test to all transport protocols and changing the existing implementations would not be possible in practice. Moreover, many transport-layer protocols either do not practice TCP-compatible congestion control or allow spoofing of acknowledgments. Therefore, the most practical solution is to do the return routability test in the IP layer.

5. DoS Attacks against BU Authentication

Security protocols that successfully protect the secrecy and integrity of data can sometimes increase the participants' vulnerability to denial-of-service attacks. In fact, the stronger – and more resource consuming – the authentication, the easier it may be for an attacker to use the protocol features to exhaust the mobile's or the correspondent's resources.

5.1. Inducing Unnecessary Authentication

When a MIPv6 mobile node receives an IP packet from a new correspondent via its home network, it may automatically send a binding update to the correspondent. The attacker can exploit this by sending the mobile spoofed IP packets (e.g. ping or TCP SYN packets) that appear to come from different correspondent addresses. The attacker will automatically start the BU protocol with all these correspondents. If the correspondent addresses are real addresses of existing IP nodes, most instances of the BU protocol will complete successfully. The entries created into the binding caches are correct but useless. This way, the attacker can induce the mobile to execute the BU protocol unnecessarily, which will drain the mobile's resources. A correspondent (i.e. any IP node) can be similarly targeted by inducing binding updates with a large number of mobiles.

This attack is possible against any BU authentication protocol. The more resources the protocol consumes, the more serious the attack. Hence, a strong cryptographic authentication protocol can be more vulnerable to the attack than a weak one or unauthenticated BUs. While we

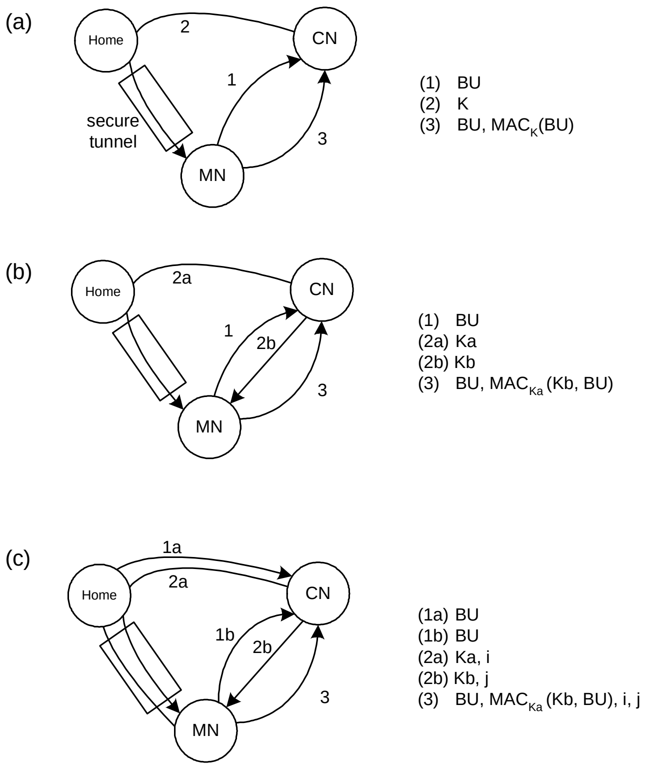

Figure 2. Evolution of the BU protocol

use MIPv6 as the case study, it is likely that similar vulnerabilities will be created by other location management schemes where the attacker can induce unnecessary location updates and authentication.

In Section 5.2, we will cover some additional DoS attacks and defense mechanisms. However, these attacks are generally no more serious than the one described in this section. It usually does not pay to defend against the other types of attacks unless we can also prevent the attack of this section.

5.2. Consuming Authentication Resources

Authentication protocols are often vulnerable to flooding attacks that exploit the protocol features to consume the target node's resources. Computing power is consumed by flooding the node with messages that cause it to perform expensive cryptographic operations. If a node creates a state during protocol execution, it is also vulnerable to attacks where the attacker starts an excessive number of protocol runs and never finishes them.

In order to exhaust the computing power of modern processors, the attacker needs to get them to perform public-key cryptographic operations. It may, for example, spoof a large number of signed messages where the signatures are replaced by random numbers. The target node will verify the signatures before rejecting the messages. Symmetric encryption, hash functions and non-cryptographic computation are rarely the performance bottleneck. However, if the cryptographic library is optimized only for bulk data, it may behave inefficiently when the functions are invoked on millions of short messages and the keys are changed on every invocation.

5.3. Reflection and Amplification

Attackers sometimes try to hide the source of a packet flooding attack by reflecting the traffic from other nodes [Pax01, Sav02]. That is, instead of sending a flood of packets directly to the target, the attacker sends data to other nodes and tricks them into sending the same number, or more, packets to the target. Reflection can hide the attacker's address even when ingress filtering prevents source address spoofing. Reflection is particularly dangerous if the packets can be reflected multiple times, if they can be sent into a looping path, or if the nodes can be tricked into sending many more packets than they receive from the attacker. Such features can be used to amplify the amount of attack traffic by a significant factor. When designing protocols, one should avoid creating services that can be used for reflection and amplification attacks.

The location management protocols could also be used for reflection. For example, the correspondent in Figure 2(b) responds to the initial packet by sending two packets to the mobile (one to the home address and one to the new address). This can be used to amplify a flooding attack by a factor of two. Furthermore, if public-key authentication is used, the packets sent by the correspondent may be significantly larger than the one that triggers them.

6. Preventing Resource Exhaustion

In this section, we discuss defenses against the DoS attacks described in the previous section. As it usually is impossible to completely prevent resource-exhaustion attacks, the right approach is to increase the cost and difficulty of the attacks and to mitigate their effects.

6.1. Delaying Commitment

A standard protection against DoS attacks is to delay committing one's resources to the protocol until the other party has provided some assurance of its honesty.

One way to avoid the unnecessary public-key operations is to require a weaker authentication, such as a routing-based method, before the expensive computation [Mea99]. This either limits the number of attackers who can get to the public-key stage or increases the cost of the attack by forcing the attacker to break the weaker mechanism first. For example, a MIPv6 binding update authentication protocol could start with a return routability test (Section 4.2) and continue with a public-key authentication only if the RR test succeeds.

Attacks that exhaust state storage can be prevented by making the protocol parties stateless [AN97] until the honesty of the other participant has been proved. While careful management of the state data can solve the problem, nodes with little memory and implementations aiming for simplicity are particularly likely to find the stateless approach easier.

There are some difficulties in making the MIPv6 binding update authentication protocol stateless. The main problem is that usually only the responder can be stateless, and it is not clear which party initiates the BU process and which one responds. Although the mobile normally initiates the authentication, this may be triggered by a packet belonging to another protocol that arrived from the correspondent via the home agent. Moreover, if a packet sent by the correspondent triggers a BU, the correspondent's IP layer does not know that this was the case because the IP layer is stateless and does not maintain a history of sent packets. For simplicity, we prefer to make the correspondent stateless until the BU has been authenticated and will not try to guess which party initiated the protocol and whether the statelessness is necessary in the particular protocol run or not.

One way in which the correspondent can remain stateless is to derive the secret values *Ka* and *Kb* of Figure 2(b) with a one-way function from a secret value N_i known only by the correspondent:

$$Ka = h(\text{ "Ka" } \mid N_i \mid \textit{mobile's home address})$$
$$Kb = h(\text{ "Kb" } \mid N_i \mid \textit{mobile's care-of address})$$

The correspondent uses the same value of N_i for all mobiles. It can discard *Ka* and *Kb* after sending the messages 2a and 2b to the mobile because it can recompute the values after receiving the final message. The correspondent generates a new secret N_i periodically, which guarantees the freshness of *Ka* and *Kb*. The index *i* helps the correspondent to identify the correct value of N_i if it happens to generate a new N_i between messages 2 and 3. Addition of this index is the only change needed to the messages of the stateful protocol.

Cryptographic puzzles [JB99, ANL00] are another proposed protection against resource-exhaustion attacks. A server requires its clients to solve a puzzle, e.g. brute-force search for some input bits of a one-way function, before committing its own resources to the protocol. The server can adjust the difficulty of the puzzles according to its load. Solving the puzzle creates a small cost for each protocol invocation, which makes flooding attacks expensive but has little effect on honest nodes. The BU protocol suggested in [MC02] relies on such puzzles for DoS protection. Unfortunately, there are several drawbacks to this strategy in location management. First, the IP layer does not know which node is the server (i.e. the respondent). Second, mobile nodes often have limited processor and battery capacity while an attacker pretending to be a mobile is likely to have much more computational resources. The puzzle protocols work well only when all clients have approximately equal processing power. Therefore, we have decided against using puzzle protocols in our design.

6.2. Limiting Damage

A node can protect itself from resource exhaustion attacks by setting a limit on the amount of resources, i.e. processor time, memory and communications bandwidth, used for location management. When the limit is exceeded, communication needs to be prioritized. For example, a MIPv6 node that exceeds the limit should stop sending or accepting BUs and allow binding cache entries to expire. Although communication can continue via the mobile's home network, it is suboptimal. The node should try to aggressively resume normal operation when it believes that the attack may be over.

Ingress filtering at the attacker's local network mitigates the resource exhaustion attacks by making it easier to trace the attacker and to filter out the unwanted packets.

6.3. Favoring Regular Customers

The correspondent's local security policy can be defined to allow BUs with some high-priority mobiles or those with which it has a long-term relationship or recent meaningful communication. The decision could be based on state information from upper protocol layers but this is problematic to implement. In some common situations, it may be worthwhile to violate the layering principle. For example, a Web server could accept BUs from its clients after it has successfully executed the TCP handshake. The mobile may similarly favor selected correspondent addresses, e.g. ones with which the mobile user has explicitly initiated communication.

It may also help to keep updating the existing entries in the Binding Cache so that existing optimized routes can be maintained during a DoS attack, although it is not certain that the existing cache entries belong to the most important mobiles or even to authentic ones. Some indication of this may be inferred from the packet counts associated with the traffic flowing through each entry.

6.4. Balancing Message Flows

Reflection attacks can be discouraged and traffic amplification prevented by ensuring that the correspondent only responds to the same address from which it received a packet, and only with a single packet of the same size. (Reflection can be prevented only if the attacker's local network applies ingress filtering.) Sometimes this can be achieved by rearranging the messages but it might be necessary to add new messages and to pad existing ones with dummy data. The question that needs to be decided is whether the cost of these protections is more acceptable than the threat created by the reflection and a small constant factor of amplification.

Figure 2(c) shows the final version of our BU-authentication protocol with one additional message to balance the message flows. Note that the correspondent can still be still stateless because it responds to message 1a with 2a and to message 1b with 2b but in no way associates the two exchanges to each other. The exchanges are parallel so that the total time taken by the protocol is not significantly increased. (*i* and *j* are the indices needed for making the correspondent stateless.) The mobile, on the other hand, needs to receive both messages 2a and 2b before sending the authenticated BU.

7. The Right Level of Protection

We conclude this paper by discussing the criteria that should be used for selecting and comparing BU authentication protocols and the issues that arise when there are several alternative protocols.

7.1. Prioritizing the Goals

It is essential to implement any protection mechanism if security of other nodes or communication between other nodes depends on it. Therefore, preventing the bombing attacks against third parties (Section 4.1) should have the highest priority when designing a secure location management protocol. In practice, this means making the return routability test (Section 4.2) mandatory. When only the node's own security and availability depends on a countermeasure, it is possible to leave the decision to each node. This is the case with most other resource-exhaustion attacks (Section 5) and, in fact, with the authentication of BU origin (Section 3). It is, however, important to realize that if a server node does not require an adequate level of authentication from its clients, the service may become unusable under attack.

In MIPv6, the binding updates are an optimization and a node can always protect itself and others by not sending BUs or by ignoring received ones. This means communicating always via the mobile's home network. This strategy can be followed when simplicity of implementation is the primary goal.

7.2. Multiple Levels of Authentication

The computational and communicational capabilities of Internet nodes vary vastly, as does the level of security they require. It would, therefore, be desirable to have a range of authentication protocols with different cost and security trade-offs. For example, closed high-security groups could use pre-established shared keys or a PKI, most nodes CGA authentication with return routability tests for DoS prevention, and low-end mobile devices a protocol based only on RR. However, care must be taken to accommodate the multiple levels of protection so that the attacker cannot bid down to the lowest level.

In MIPv6, the decision about accepting or rejecting a BU is made by the correspondent. Therefore, the correspondent will always make the final decision about the required level of authentication for a particular mobile. It makes little sense for the correspondent to allow multiple levels of authentication for the same mobile node because the attacker could always tackle the weakest one. Thus, the mobile must either authenticate itself using the protocol chosen by the correspondent or give up binding updates. Protocol negotiation is counterproductive unless the mobile's choices can be strongly authenticated.

A technique similar to CGA addresses can be used to express the mobile's choice. The idea is to hash the list of acceptable protocols together with the mobile's public key and routing prefix when forming the interface identifier (i.e. the second half of the IP address). An alternative proposal is to reserve type bits in the IP address to indicate whether the address is cryptographically generated or of some other type. Both techniques are based on the observation that if the mobile's choice of protocol is encoded into the IP address, the attacker cannot interfere with it.

It is worth noting that as long as bombing of third parties is prevented, different correspondents can make their choice of authentication strength independently. This is because a weak mechanism accepted by one correspondent will not help the attacker to redirect packets to or from correspondents that use a stronger protocol. The correspondent can also have a local policy that mandates a stronger (e.g. shared key authentication or PKI) or weaker (e.g. plain RR) of authentication for a particular home address or range of addresses.

There is, however, the risk that business reasons will force practically all IP nodes to use the weakest level of authentication that is mandatory to implement and use. For example, if many low-end mobiles only implement the weakest standardized protocol, virtually all correspondents will default to this mechanism, which would defeat the purpose of having any stronger protocol.

8. Conclusions

We described attacks against Internet location management protocols with particular focus on Mobile IPv6 binding updates. Some of the attacks are new in the sense that before our threat analysis they had not been considered in the IETF Mobile IP Working Group. In particular, the flooding attack against third parties (Section 4.1) has been ignored in many Internet protocols that update location or routing information. We also suggested and analyzed mechanisms for protecting the protocol participants and third parties. The ideas presented in this paper formed the basis for the development of a secure location management protocol for Mobile IPv6, which uses only symmetric cryptography and follows the lines of Figure 2(c). We hope that this work will help to secure other Internet mobility protocols as well.

Acknowledgments

Many of the ideas were influenced by Greg O'Shea, Pekka Nikander, Erik Nordmark, Gabriel Montenegro, and various Internet Drafts.

References

[AN97] Tuomas Aura and Pekka Nikander. Stateless connections. In *Proc. International Conference on Information and Communications Security (ICICS'97)*, volume 1334 of LNCS, pages 87-97, Beijing, China, November 1997. Springer.

[ANL00] Tuomas Aura, Pekka Nikander, and Jussipekka Leiwo. DOS-resistant authentication with client puzzles. In *Proc. Security Protocols Workshop 2000*, volume 2133 of LNCS, pages 170-181, Cambridge, UK, April 2000. Springer.

[Eas97] Donald E. Eastlake 3rd. Secure domain name system dynamic update. RFC 2137, IETF Network Working Group, April 1997.

[FS00] Paul Ferguson and Daniel Senie. Network ingress filtering: Defeating denial of service attacks which employ IP source address spoofing. RFC 2827, IETF Network Working Group, May 2000.

[HD98] Robert M. Hinden and Stephen E. Deering. IP version 6 addressing architecture. RFC 2373, IETF Network Working Group, July 1998.

[JPA02] David B. Johnson, Charles Perkins, and Jari Arkko. Mobility support in IPv6. Internet-Draft draft-ietf-mobileip-ipv6-18.txt, IETF Mobile IP Working Group, June 2002. Work in progress.

[JB99] Ari Juels and John Brainard. Client puzzles: a cryptographic countermeasure against connection depletion attacks. In *Proc. 1999 Network and Distributed Systems Security Symposium (NDSS)*, pages 151-165, San Diego, CA USA, February 1999. Internet Society.

[KS99] Phil Karn and William A. Simpson. Photuris: session-key management protocol. RFC 2522, IETF Network Working Group, March 1999.

[KA98] Stephen Kent and Randall Atkinson. Security architecture for the Internet Protocol. RFC 2401, IETF Network Working Group, November 1998.

[Mea99] Catherine Meadows. A formal framework and evaluation method for network denial of service. In *Proc. 12th IEEE Computer Security Foundations Workshop*, pages 4-13, Mordano, Italy, June 1999. IEEE Computer Society.

[MC02] Gabriel Montenegro and Claude Castelluccia. Statistically unique and cryptographically verifiable identifiers and addresses. In *Proc. ISOC Symposium on Network and Distributed System Security (NDSS 2002)*, San Diego, February 2002.

[ND01] Thomas Narten and Richard Draves. Privacy extensions for stateless address autoconfiguration in IPv6. RFC 3041, IETF Network Working Group, January 2001.

[NNS98] Thomas Narten, Erik Nordmark, and William Allen Simpson. Neighbor discovery for IP version 6 (IPv6). RFC 2461, IETF Network Working Group, December 1998.

[Nik01] Pekka Nikander. A scaleable architecture for IPv6 address ownership. Internet-draft, March 2001. Work in Progress.

[OR01] Greg O'Shea and Michael Roe. Child-proof authentication for MIPv6 (CAM). *ACM Computer Communications Review*, 31(2), April 2001.

[Pax01] Vern Paxson, An analysis of using reflectors for distributed denial-of-service attacks. *ACM Computer Communication Review*, 31(3), July 2001.

[PJ01] Charles Perkins and David B. Johnson. Route optimization in mobile IP. Internet-Draft draft-ietf-mobileip-optim-11.txt, IETF Mobile IP Working Group, September 2001. Work in progress.

[RAOA02] Michael Roe, Tuomas Aura, Greg O'Shea, and Jari Arkko. Authentication of mobile IPv6 binding updates and acknowledgments. Internet-Draft draft-roe-mobileip-updateauth-02.txt, IETF Mobile IP Working Group, February 2002. Work in progress.

[Sav02] Pekka Savola. Security of IPv6 routing header and home address options. Internet-draft, IETF, November 2002. Work in progress.

[SKK+97] Christoph L. Schuba, Ivan V. Krsul, Markus G. Kuhn, Eugene H. Spaffold, Aurobindo Sundaram, and Diego Zamboni. Analysis of a denial of service attack on TCP. In *Proc. 1997 IEEE Symposium on Security and Privacy*, pages 208-223, Oakland, CA USA, May 1997. IEEE Computer Society Press.

[TN98] Susan Thomson and Thomas Narten. IPv6 stateless address autoconfiguration. RFC 2462, IETF Network Working Group, December 1998.

[VTRB97] Paul Vixie, Susan Thomson, Yakov Rekhter, and Jim Bound. Dynamic updates in the domain name system (DNS UPDATE). RFC 2136, IETF Network Working Group, April 1997.